

Computation of Extremum Singular Values and the Strong H-infinity Norm of SISO Time-Delay Systems

Suat Gumussoy^a, Wim Michiels^b

^a*Math Works,
3 Apple Hill Drive, Natick, MA 01760, USA
(e-mail: suat.gumussoy@mathworks.com).*

^b*Department of Computer Science, KU Leuven,
Celestijnenlaan 200A, 3001, Heverlee, Belgium
(e-mail: wim.michiels@cs.kuleuven.be).*

Abstract

We consider the computation of H-infinity norms for Single-Input-Single-Output (SISO) time-delay systems, which are described by delay differential algebraic equations. Unlike the iterative level set methods in the literature, we present a novel numerical method to compute the H-infinity norm. This method requires solving one eigenvalue problem of at most twice the size of the eigenvalue problem in every iteration of a level set method, but in practice often considerably lower. We first show that the computation of extrema of the transfer function can be turned into the computation of the imaginary axis zeros of a transcendental function. We compute these zeros by a predictor-corrector type algorithm. It is known that the H-infinity norm of delay differential algebraic systems, which can model both retarded and neutral type systems, might be sensitive with respect to arbitrarily small delay perturbations. This recently led to the concept of strong H-infinity norms, which explicitly take into account such small delay perturbations. We present a direct numerical method to compute the strong H-infinity norm of SISO time-delay systems. Our algorithm is applicable to the closed-loop system of interconnections (series, parallel, feedback, junctions) of time-delay systems and/or controllers.

Key words: time-delay system, robust control, H-infinity norm, computational methods

1 Introduction

The availability of robust methods to compute \mathcal{H}_∞ norms is essential in a computer aided control system design [15]. The common approaches for computing \mathcal{H}_∞ norms for finite-dimensional linear systems, which belong to the class of level set methods, are based on the relation between the intersections of singular value curves of the transfer function with a constant function (the level) and the presence of imaginary axis eigenvalues of a corresponding Hamiltonian matrix [6]. In [6,2] linearly converging bisection based algorithms have been proposed. Quadratically convergent algorithms, relying on an alternating search in two directions, have been described in [1,5]. In [12] a level set algorithm for computing \mathcal{H}_∞ norms of a class of retarded type time-delay systems has been outlined. The main complication with respect to the delay-free case is that the intersections between the singular value curves and a level set are no longer related to the spectrum of a Hamiltonian matrix, but to an infinite-dimensional operator. Therefore, the

algorithm of [12] adopts a predictor-corrector approach, where in the first step the operator is discretized using a spectral method, followed by local corrections to remove the effect of the discretization on computed peak values in the frequency response. For the sake of completeness, it should be mentioned that for systems without delay, \mathcal{H}_∞ norms can also be computed by a parameter sweep, thereby checking the feasibility of linear matrix inequalities, see, e.g., [13]. However, since these methods implicitly construct a Lyapunov function, and in the infinite-dimensional, time-delay case, fixing the form of the Lyapunov functional to a tractable form involving finitely many free parameters typically introduces conservatism, only potentially conservative bounds on the \mathcal{H}_∞ norm can be aimed at in the latter case.

Inherent to level set methods, whether for systems with or without delay, is that the \mathcal{H}_∞ norm is computed in an iterative way, by updating the level in every iteration step until convergence to the dominant peak in the singular value plot is achieved. In every iteration step the

imaginary axis eigenvalues of a Hamiltonian matrix or operator need to be computed. As a first main contribution of the paper, we present a novel numerical algorithm to compute \mathcal{H}_∞ norms of SISO time-delay systems requiring solving only *one* eigenvalue problem.

The second main contribution is that our method admits a system description in a *standard form*, described by a set of delay differential algebraic equations (DDAEs). As we will see, this form contains a large set of systems, including interconnections of time-delay systems and controllers in complex configurations, and including both retarded and neutral type systems.

Recently, we analyzed in [9] the properties of the \mathcal{H}_∞ norm of time-delay systems. We illustrated that the \mathcal{H}_∞ norm of DDAEs may be sensitive with respect to arbitrarily small delay perturbations. Due to this sensitivity, we introduced the *strong \mathcal{H}_∞ norm*, which explicitly takes into account small delay perturbations, inevitable in any practical control application, and we outlined the computation using a level set approach. The derived theory of strong \mathcal{H}_∞ norms can be considered as the dual of the theory of strong stability for neutral systems and DDAEs, as, e.g., elaborated in [10,11] and the references therein. As a third, main contribution, the presented algorithm in this paper takes the potential sensitivity problem into account, by computing the strong \mathcal{H}_∞ norm. In control problems without feed through at infinity along the loops, this strong \mathcal{H}_∞ norms reduces to the standard \mathcal{H}_∞ norm.

The remainder of the paper organizes as follows. Section 2 describes the standard form on time-delay systems on which the algorithms rely. Section 3 presents the computation of extrema of the transfer function of a SISO time-delay system. In particular, §3 shows that the local maximum and minimum can be computed by finding the imaginary axis zeros of a transcendental function. In §3.1 these zeros are computed by a predictor-corrector algorithm. Based on this computation and a characterization of the high-frequency behavior, we present a non-iterative numerical algorithm for the strong \mathcal{H}_∞ norm computation in Section 4. Numerical examples and concluding remarks are given in Sections 5 and 6.

Notation: The sets of complex, real and integer numbers are $\mathbb{C}, \mathbb{R}, \mathbb{Z}$ and the set of strictly positive and nonnegative real numbers are $\mathbb{R}^+, \mathbb{R}_0^+$. Zero and identity matrices are $0, I$. The transpose of the matrix A is A^T . Complex conjugate transpose, i^{th} maximum singular value and derivative of transfer function $G(s)$ are shown as $G^*(s)$, $\sigma_i(G)$, $G'(s)$ respectively. The function constructing block diagonal matrix from input arguments is `blkdiag`.

2 Standard form for time-delay systems

We consider a general representation for a time-delay system described by the delay differential algebraic equations,

$$G : \begin{cases} E\dot{\bar{x}}(t) = A_0\bar{x}(t) + \sum_{i=1}^m A_i\bar{x}(t - \tau_i) + Bu(t), \\ y(t) = C\bar{x}(t), \end{cases} \quad (1)$$

where $E, A_i \in \mathbb{R}^{n \times n}$ for $i = 0, \dots, m$, $B \in \mathbb{R}^{n \times n_u}$ and $C \in \mathbb{R}^{n_y \times n}$ are real valued matrices with appropriate dimensions. The time-delays τ_i for $i = 0, \dots, m$ are non-negative real numbers. Let matrix E in (1) satisfy

$$\text{rank}(E) = n - \nu,$$

with $0 \leq \nu < n$. In case $\nu \geq 1$, i.e., matrix E is singular, we let the columns of matrix $U \in \mathbb{R}^{n \times \nu}$, respectively $V \in \mathbb{R}^{n \times \nu}$, be a (minimal) orthonormal basis for the left, respectively right nullspace of E , which implies $U^T E = 0$, $EV = 0$. We then make the following assumption.

Assumption 1 *Matrix $U^T A_0 V$ is nonsingular.*

This assumption is necessary to make system (1) causal [11]. We refer to this reference for a discussion of basic properties of (1) (definition of solutions, spectrum determined growth property of solutions, ...).

Assumption 2 *The null solution of system (1), with $u \equiv 0$, is strongly exponentially stable.*

The stability of the time-delay system (1) is a necessary assumption for \mathcal{H}_∞ norm computation since this norm is finite for stable systems only. Strong exponential stability refers to the fact that the asymptotic stability of the null solution is robust against small delay perturbations, [10].

The representation in (1) is the standard form of the SISO time-delay plant for our algorithms. This form is *closed* under block diagram operations such as series, parallel, feedback etc. and therefore, it is rich enough to represent most of the systems as a result of interconnections of time-delay systems and/or controllers. This form (1) allows to represent systems with multiple state, multiple input and output delays, as well as systems with a nontrivial feed through including neutral type systems, see [9]. This generality is an important property since the \mathcal{H}_∞ norm computation of the *closed-loop* system with the designed controller is a common scenario. Therefore representing the closed-loop system and all its subsystems with the same form is essential for practical use of the algorithm.

Another representation for the time-delay systems (also used in MATLAB) is the *generalized LTI* (GLTI) class

of continuous-time LTI systems, where systems are modeled as the LFT interconnection of a delay-free LTI model H and a set of internal, input and output delays (see Figure 1).

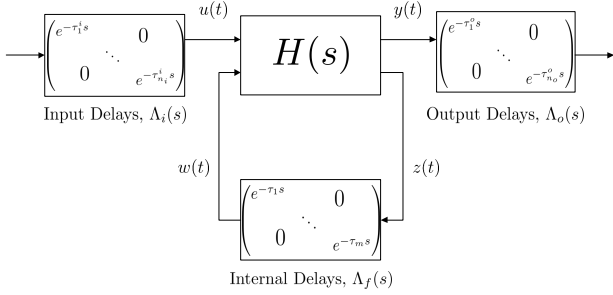


Fig. 1. Systems with internal, input and output delays

The class of GLTI systems is also closed under series, parallel, and feedback connections as well as branching/summing junctions [7].

State-space equations for $H(s)$ and time-delay terms in Figure 1 are

$$\begin{aligned} F\dot{x}(t) &= Ax(t) + B_1u(t) + B_2w(t) \\ y(t) &= C_1x(t) + D_{11}u(t) + D_{12}w(t) \\ z(t) &= C_2x(t) + D_{21}u(t) + D_{22}w(t) \\ w(t) &= (\Lambda_f z)(t) \end{aligned} \quad (2)$$

where $\Lambda_f z$ is the vector-valued signal defined by $(\Lambda_f z)(t) := (z_1^T(t - \tau_1), \dots, z_m^T(t - \tau_m))^T$. Λ_i and Λ_o operate on signals similarly, see [8] for further details on defining various types of systems with time-delays. We can transform the time-delay system in LFT form (2) into our standard form (1) by first defining the augmented state $\bar{x}^T := [x^T \ \gamma_u^T \ w^T \ z^T]^T$ where γ_u is the auxiliary variable for the input signal u . Then we rewrite system equations (2) in terms of the new state variable \bar{x} as in (1)

$$\begin{aligned} E &= \text{blkdiag}(F, 0, 0, 0), \\ A_0 &= \begin{bmatrix} A & B_1 & B_2 & 0 \\ C_2 & D_{21} & D_{22} & -I \\ 0 & 0 & -I & 0 \\ 0 & -I & 0 & 0 \end{bmatrix}, A_i = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & I \\ 0 & 0 & 0 & 0 \end{bmatrix} \\ B &= \begin{bmatrix} 0 & 0 & 0 & I \end{bmatrix}^T, C = \begin{bmatrix} C_1 & D_{11} & D_{12} & 0 \end{bmatrix}. \end{aligned}$$

Hence, the transformation of (2) to (1) is immediate.

In the remainder of the paper we assume that, $n_u = n_y = 1$, i.e., (1) is a SISO system.

3 Computation of extremum singular values

The transfer function of (1) is equal to

$$G(s) = C(sE - A_0 - \sum_{i=1}^m A_i e^{-s\tau_i})^{-1} B. \quad (3)$$

In what follows we characterize local extrema of the function

$$\mathbb{R} \ni \omega \mapsto |G(j\omega)|. \quad (4)$$

An extremum of the singular value curve of $G(j\omega)$ satisfies

$$\begin{aligned} 0 &= \frac{d}{d\omega} G(j\omega)G(j\omega)^* = \frac{d}{d\omega} G(j\omega)G(-j\omega) \\ &= jG'(j\omega)G(-j\omega) - jG(j\omega)G'(-j\omega). \end{aligned}$$

We arrive at the following result.

Proposition 1 *The extrema of (4) and corresponding frequencies can be obtained from the imaginary axis zeros of the transcendental function*

$$Z(s) := G'(s)G(-s) - G'(-s)G(s). \quad (5)$$

We introduce the following notation and two lemmas to derive a state-space like representation (5). Let $\mathbb{A}(s) = A_0 + \sum_{i=1}^m A_i e^{-s\tau_i}$ and let \mathbb{A}' be the derivative of \mathbb{A} with respect to s . Then, we can write the transfer function of the SISO time-delay system (3) as

$$G(s) = C(sE - \mathbb{A}(s))^{-1} B. \quad (6)$$

Lemma 2 *Let $G(s)$ have a transfer function (6). The derivative of $G(s)$ with respect to s is*

$$G'(s) = [C \ 0] \left(s \begin{bmatrix} E & 0 \\ 0 & E \end{bmatrix} - \begin{bmatrix} \mathbb{A} & \mathbb{A}' - E \\ 0 & \mathbb{A} \end{bmatrix} \right)^{-1} \begin{bmatrix} 0 \\ B \end{bmatrix}.$$

Proof. A simple computation yields

$$\begin{aligned} G'(s) &= -C(sE - \mathbb{A})^{-1}(E - \mathbb{A}')(sE - \mathbb{A})^{-1}B, \\ &= [C \ 0] \begin{bmatrix} sE - \mathbb{A} & E - \mathbb{A}' \\ 0 & sE - \mathbb{A} \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ B \end{bmatrix}. \quad \square \end{aligned}$$

Using Lemma 2, we can derive the state space representation for $G'(s)$ and $G'(-s)$. The state space representation of (5) can be derived by the following result, which is a natural extension of the finite-dimensional case.

Lemma 3 Let $G_1(s) = C_1(sE_1 - \mathbb{A}_1)^{-1}B_1$ and $G_2(s) = C_2(sE_2 - \mathbb{A}_2)^{-1}B_2$. Then

$$G_1(s) + G_2(s) = [C_1 \ C_2] \begin{bmatrix} sE_1 - \mathbb{A}_1 & 0 \\ 0 & sE_2 - \mathbb{A}_2 \end{bmatrix}^{-1} \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}$$

and

$$G_2(s)G_1(s) = [0 \ C_2] \begin{bmatrix} sE_1 - \mathbb{A}_1 & 0 \\ -B_2C_1 & sE_2 - \mathbb{A}_2 \end{bmatrix}^{-1} \begin{bmatrix} B_1 \\ 0 \end{bmatrix}.$$

By Lemma 2 and 3, we arrive at the following representation of $Z(s)$ in (5):

Proposition 4 We can express

$$Z(s) = C_z(sE_z - \mathbb{A}_z(s))^{-1}B_z, \quad (7)$$

where the matrices are given by

$$\begin{aligned} E_z &= \text{blkdiag}(-E, E, E, -E, E, E), \\ \mathbb{A}_z(s) &= \text{blkdiag}(\mathcal{A}_z(s), \mathcal{A}_z(-s)), \\ B_z &= [B^T \ 0 \ 0 \ B^T \ 0 \ 0]^T, \quad C_z = [0 \ C \ 0 \ 0 \ C \ 0], \end{aligned}$$

$$\text{and } \mathcal{A}_z(s) = \begin{bmatrix} \mathbb{A}(-s) & 0 & 0 \\ 0 & \mathbb{A}(s) & \mathbb{A}'(s) - E \\ BC & 0 & \mathbb{A}(s) \end{bmatrix} \in \mathbb{R}^{3n \times 3n}.$$

The next section presents the numerical algorithm to compute the zeros of Z using the above representation.

3.1 Computing the zeros of Z

Based on (7) we can compute the imaginary axis zeros of Z in (5) by computing the imaginary axis solutions of the following nonlinear eigenvalue problem

$$\begin{bmatrix} \mathbb{A}_z(s) - sE_z & B_z \\ C_z & 0 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = 0, \quad (8)$$

where $[u^T, v^T]^T$ is the corresponding eigenvector.

The nonlinearity of the eigenvalue problem stems from the fact that $\mathbb{A}_z(s)$ depends on $e^{-s\tau_i}$ and $e^{s\tau_i}$, $i = 1, \dots, m$. This makes that the number of solutions of (8) is in general infinite. It is important to note that the solutions are symmetric with respect to the imaginary axis, as can easily be seen from (5). Therefore, solutions either appear in quadruples $(s, \bar{s}, -s, -\bar{s})$, or in pairs on

the imaginary axis, the latter corresponding to the extrema of the original transfer function.

Due to the nonlinearity of (8), an approximation is necessary to globally detect zeros on the imaginary axis. This brings us to a predictor-corrector approach to solve the problem, inspired by [9]. For the predictor step, we start from a rational approximation of (3):

$$G_N(\lambda) := \mathbf{C}_N(\lambda \mathbf{E}_N - \mathbf{A}_N)^{-1} \mathbf{B}_N, \quad (9)$$

obtained by a spectral discretization of the delay system on a grid of N Chebyshev points [4,9]. See Appendix B in [9] for the computation of the system matrices in (9). Subsequently, we determine the extrema of the curve $\omega \mapsto G_N(j\omega)$. Similarly to the above derivation, these are given by the imaginary axis zeros of

$$Z_N(s) := G'_N(s)G_N(-s) - G'_N(-s)G_N(s). \quad (10)$$

Let G_N has the zero-pole-gain form as $G_N(s) = k \frac{b(s)}{a(s)}$. Then, we can write Z_N as

$$Z_N(s) = \frac{\overbrace{\left(-\frac{b'(-s)}{b(-s)} + \frac{b'(s)}{b(s)} + \frac{a'(-s)}{a(-s)} - \frac{a'(s)}{a(s)} \right)}^{\Delta(s)}}{k^2 \overbrace{a^3(s)a^3(-s)b(s)b(-s)}^{\Delta(s)}}.$$

Note that the imaginary axis zeros as Z_N and Δ are the same and we can compute the system Δ as

$$\Delta(s) = \sum_{i=1}^{n_z} \left(\frac{1}{s + \hat{z}_i^*} + \frac{1}{s - \hat{z}_i} \right) - \sum_{k=1}^{n_p} \left(\frac{1}{s + \hat{p}_k^*} + \frac{1}{s - \hat{p}_k} \right)$$

where \hat{z}_i and \hat{p}_k are zeros and poles of the system G_N . The computation of the imaginary axis zeros of Δ requires to solve the (standard) generalized eigenvalue problem of size $2(n_p + n_z) + 1$,

$$\begin{bmatrix} A_\Delta - sI & B_\Delta \\ C_\Delta & 0 \end{bmatrix} \begin{bmatrix} u_N \\ v_N \end{bmatrix} = 0, \quad (11)$$

where A_Δ , B_Δ and C_Δ are system matrices of the system Δ . Due to the singularity of matrix E in interconnected system, the computational cost could be significantly lower than the order of system G_N , $n(N+1)$.

Because of the error induced by replacing $G(s)$ with $G_N(s)$, the imaginary axis eigenvalues of (11) will only be approximations of the imaginary axis eigenvalues of (8) looked for. Therefore, the second, correction step of the algorithm serves to remove the discretization error. It is based on solving a system of nonlinear equations

that characterize extremum points in the singular value curve of G , where the initial values are obtained in the first, predictor step. These equations are given by

$$H(j\omega, \xi) \begin{bmatrix} u \\ v \end{bmatrix} = 0, \quad \Im\{v^*(E_z - A'_z(j\omega))u\} = 0, \quad (12)$$

and the normalization constraints $n(u, v) = 0$ where

$$H(j\omega, \xi) := \begin{bmatrix} j\omega E - A(j\omega) & -\xi^{-1}BB^T \\ \xi^{-1}C^TC & j\omega E^T + A^T(-j\omega) \end{bmatrix}.$$

The first equation in (12) expresses that ξ is a singular value of G at frequency ω by rewriting the singular value equation in the form of a Hamiltonian eigenvalue problem. The second equation expresses that the Hamiltonian eigenvalue problem has a double imaginary axis solution. This property corresponds to ξ being an extremum of the singular value curve and is equivalent to a zero derivative of the singular value curve with respect to the ω . Finally normalization constraints for the singular vectors need to be added to make the solution unique. For more details we refer to [12].

Remark 5 *The reason for solving (12) in the correction step instead of solving (8) is that the number of equations is smaller ($4n + 3$ instead of $12n + 2$ real equations).*

The overall algorithm for the computation of extremum singular values is as follows.

Algorithm 1 Input: system data, N , grid Ω_N with grid points (see [9].)

(1) Prediction step:

- (a) Compute all imaginary axis zeros of Z_N in (10) by computing the generalized eigenvalues of the pencil (11), whose imaginary axis eigenvalues are given by $\lambda = j\omega^{(i)}$.
- (b) For each imaginary axis eigenvalue, compute the predicted extremum point $\tilde{\xi}^{(i)}$ of the singular value curve of G in (1) by $\tilde{\xi}^{(i)} = |G(j\tilde{\omega}^{(i)})|$.

(2) Correction step:

Solve the nonlinear equations (12) using the Gauss-Newton method, with the starting values

$$\omega = \tilde{\omega}^{(i)}, \quad \begin{bmatrix} u \\ v \end{bmatrix} = \arg \min_{\xi} \|H(j\tilde{\omega}^{(i)}, \tilde{\xi}^{(i)})\zeta\| / \|\zeta\|;$$

denote the solutions with $(\hat{u}^{(i)}, \hat{v}^{(i)}, \hat{\omega}^{(i)}, \hat{\xi}^{(i)})$, for $i = 1, 2, \dots$

- (3) The extremum singular values of the time-delay system G in (1) and their frequencies are $(\hat{\xi}^{(i)}, \hat{\omega}^{(i)})$ for $i = 1, 2, \dots$

There is a linear relationship with the number N and the length of the frequency range approximated. The cut-off frequency as a function of N is illustrated in Figure 4.1 of [12]. Experience from extensive benchmarking learns that in most practical problems a very small value of N can be taken (the default $N = 20$ is largely sufficient). For further details of the choice of the number of discretization points, N , we refer to [12], whose approach extends to the problem considered.

4 Direct computation of the strong \mathcal{H}_∞ norm

The \mathcal{H}_∞ norm of an asymptotically stable SISO system with transfer function (3) satisfies

$$\|G\|_\infty := \sup_{\omega \geq 0} |G(j\omega)|. \quad (13)$$

Algorithm 1 computes finite extrema of the transfer function. This is not sufficient for the following reasons.

- (1) Description (1) allows to model systems with a non-trivial feed through. As a consequence, the \mathcal{H}_∞ norm might not be reached at a finite frequency.
- (2) It is shown in [12] that the standard \mathcal{H}_∞ norm of linear delay-differential algebraic systems (and neutral type systems) might be sensitive to infinitesimal perturbations of the time-delays. The sensitivity takes, for instance, place in control loops which have a feed trough at infinity, prone to time-delays. It is due to the high frequency behavior of the transfer function (3), which is described by the *asymptotic transfer function* $G_a(s)$ defined as

$$-CV(U^T A_0 V + \sum_{i=1}^m U^T A_i V e^{-s\tau_i})^{-1} U^T B. \quad (14)$$

The above two observations have led in [9] to the introduction of the concept of *strong* \mathcal{H}_∞ norm, which is the smallest upper bound robust against infinitesimal delay perturbations. Making the dependence of G on the delays $\vec{\tau} = (\tau_1, \dots, \tau_m)$ explicit with the notation $G(\lambda; \vec{\tau})$, we have:

Definition 6 For given delays $\vec{\tau} \in (\mathbb{R}_0^+)^m$, the strong \mathcal{H}_∞ norm of G , $\|G\|_\infty$, is defined as

$$\lim_{\epsilon \rightarrow 0^+} \sup \{ \|G(j\omega; \vec{\tau}_\epsilon)\|_\infty : \vec{\tau}_\epsilon \in \mathcal{B}(\vec{\tau}, \epsilon) \cap (\mathbb{R}^+)^m \}.$$

where $\mathcal{B}(\vec{\tau}, \epsilon)$ is open ball of radius ϵ centered at $\vec{\tau} \in (\mathbb{R}^+)^m$, $\mathcal{B}(\vec{\tau}, \epsilon) := \{\vec{\theta} \in (\mathbb{R}^+)^m : \|\vec{\theta} - \vec{\tau}\| < \epsilon\}$.

The definition of $\|G_a\|_\infty$ is analogous. The following results can be found in [9].

Proposition 7 *The assertions below hold.*

- The asymptotic transfer function satisfies

$$\|G_a\|_\infty = \max_{\tilde{\theta} \in [0, 2\pi]^m} |CV(U^T A_0 V + \sum_{i=1}^m U^T A_i V e^{j\theta_i})^{-1} U^T B| \quad (15)$$

with the argument of the max operator continuous in $\tilde{\theta}$.

- The strong \mathcal{H}_∞ norm of G is equal to

$$\|G\|_\infty = \max(\|G\|_\infty, \|G_a\|_\infty). \quad (16)$$

The first assertion allows a computation of $\|G_a\|_\infty$ by gridding in the $\tilde{\theta}$ space. It should be stressed that in most applications the number of actual time-delays appearing in G_a is much smaller than the number of system delays, m , reducing significantly the computational cost. This is because most of the terms in the parenthesis of (14) are typically zero. The nonzero terms correspond to a high frequency feed through paths over the control loop.

From the second assertion it follows that if $\|G\|_\infty > \|G_a\|_\infty$, the (strong) \mathcal{H}_∞ norm of G is reached at a finite frequency. Hence, a combination of computing $\|G_a\|_\infty$ with Algorithm 1 allows to compute $\|G\|_\infty$. We arrive at Algorithm 2.

Algorithm 2 Input: system data.

- (1) The strong \mathcal{H}_∞ norm of G_a :
Compute ξ_s^a , the strong \mathcal{H}_∞ norm of the asymptotic transfer function G_a , by gridding.
- (2) The strong \mathcal{H}_∞ norm of G :
By Algorithm 1, compute (ξ_o, ω_o) , the maximum of $|G(j\omega)|$ in (1) and its frequency, where $\xi_o = \hat{\xi}^{(i_o)} = \max(\hat{\xi}^{(1)}, \hat{\xi}^{(2)}, \dots)$ and $\omega_o = \hat{\omega}^{(i_o)}$.
Compute ξ_s , the strong \mathcal{H}_∞ norm of G , where $\xi_s = \max(\xi_s^a, \xi_o)$, and its frequency ω_s , where $\omega_s = \omega_o$ if $\xi_o > \xi_s$ or $\omega_s = \infty$ otherwise.

There are two parts in the computation cost of the strong \mathcal{H}_∞ norm. The first part is to find the strong \mathcal{H}_∞ norm of the asymptotic transfer function G_a . As pointed out the number of delays appearing in G_a is usually much smaller than the number of system delays. Therefore the computation cost for the first step is not usually high. The second main part is the computation of the generalized eigenvalues of pencil (11) in the prediction step of Algorithm 1 with dimensions $2(n_p + n_z) + 1$, where n_p and n_z are the poles and zeros of the descriptor system G_N with order $n(N + 1)$. The default value for N is 20 in our code. Inherent to the DAE modeling framework, matrix E is often singular, leading values of n_z and n_p considerably lower than the order G_N (see the next section for an example). The algorithm for \mathcal{H}_∞ norm computation in [12] only applies to retarded time-delay systems. The \mathcal{H}_∞ norm computation in [12] is iterative due

to the level set approach and requires solving an eigenvalue problem of size $2n(N + 1)$ for each level set. On the other hand, the algorithm in [9] considers retarded and neutral type time-delay systems and has the same numerical cost as Algorithm 2 in the first part and in the second part it solves a generalized eigenvalue problem with dimensions $2n(N + 1)$ in every iterative step.

5 Numerical Examples

We consider the Smith Predictor example in [8] where the subsystems are $P = 6e^{-106s}/(37s + 1)$, $G_p = 5.6/(40.2s + 1)$, $D_p = e^{-93.9s}$, $C = 0.5(1 + 1/(40 * s))$ and $F = 1/(20s + 1)$.

The closed-loop system T_{sp} is a generalized LTI with internal delays and its Bode magnitude plot is shown in Figure 2. Circles indicate the computed imaginary axis zeros of $Z(s)$ (5) in the prediction step, for $N = 20$. Dots show the results after the prediction results are corrected, inducing a move to the extremum locations of the singular value curve. Note that the predicted extremum points do not improve much after correction step for initial points since the approximation in the prediction step reliably computes the points. As shown in Figure 2 at the bottom, zoomed to larger frequencies, we see the improvement in the correction step due to slight deviation of the approximation from the exact values. The closed-loop system is a retarded time-delay system and its asymptotic transfer function is equal to zero which can be seen from the high frequency behavior in Figure 2. Therefore, the strong \mathcal{H}_∞ norm of T_{sp} is equal to the standard \mathcal{H}_∞ norm of T_{sp} by (16). The norm is equal to the largest singular value 1.3308 of 4th point in the extremum points. This example has a singular E matrix, therefore, the algorithm in [12] is not applicable. When we compared the prediction steps of our algorithm (including pole, zero computation of G_N) and the one in [9], the computation time is 0.1885 and 0.6447 seconds on Intel Xeon 3.06GHz with 12GB RAM. The size of the (single) eigenvalue problem for our algorithm is 175, whereas the size of the eigenvalue problem in [9] is $2n(N + 1) = 378$ (to be solved in every iteration).

The second example considers the case where the strong \mathcal{H}_∞ norm, computed using Algorithm 2, is different from the standard \mathcal{H}_∞ norm. Given $\tau_1 = 1$, $\tau_2 = 2$, the transfer function T_{sh} is equal to

$$T_{sh} = (s + 2.1)/((s + 0.1)(1 - 0.25e^{-\tau_1 s} + 0.5e^{-\tau_2 s}) + 1).$$

The magnitude plot of T_{sh} is shown in Figure 3 on the top and the extremum points are marked as above. Note that the high frequency behavior in Figure 3 on the top does not converge to zero. Therefore, the asymptotic transfer function is different from zero. As a first step of the algorithm, the strong \mathcal{H}_∞ norm of the asymptotic

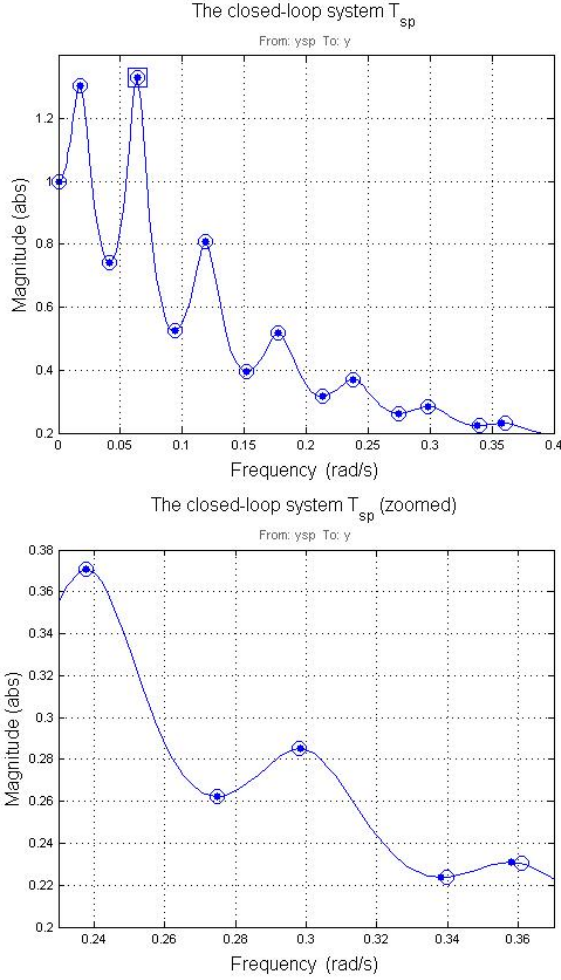


Fig. 2. (top) The magnitude Bode plot of T_{sp} and its extremum points computed at prediction and correction steps (bottom), the zoomed version with last five extremum points.

transfer function T_a , given by

$$T_a = 1/(1 - 0.25e^{-\tau_1 s} + 0.5e^{-\tau_2 s}),$$

is computed and equals 4, obtained at $\theta_1 = 0$ and $\theta_2 = \pi$ in formula (15).

In the second step, we compute the extrema of the magnitude plot by Algorithm 1. The standard \mathcal{H}_∞ norm of T_{sh} is equal to 2.5788 and the high frequency behavior visualized in Figure 3 on the top. Therefore, the strong \mathcal{H}_∞ norm of T_{sh} is equal to 4 by the final step of Algorithm 2, which is larger than the standard \mathcal{H}_∞ norm. This illustrates that the \mathcal{H}_∞ norm may be sensitive to small delay changes. Figure 3 at the bottom shows that the strong \mathcal{H}_∞ norm value 4 is achieved for a slight perturbation in the delay τ_1 and it can be shown that this norm is attained a larger frequency for smaller delay perturbation size.

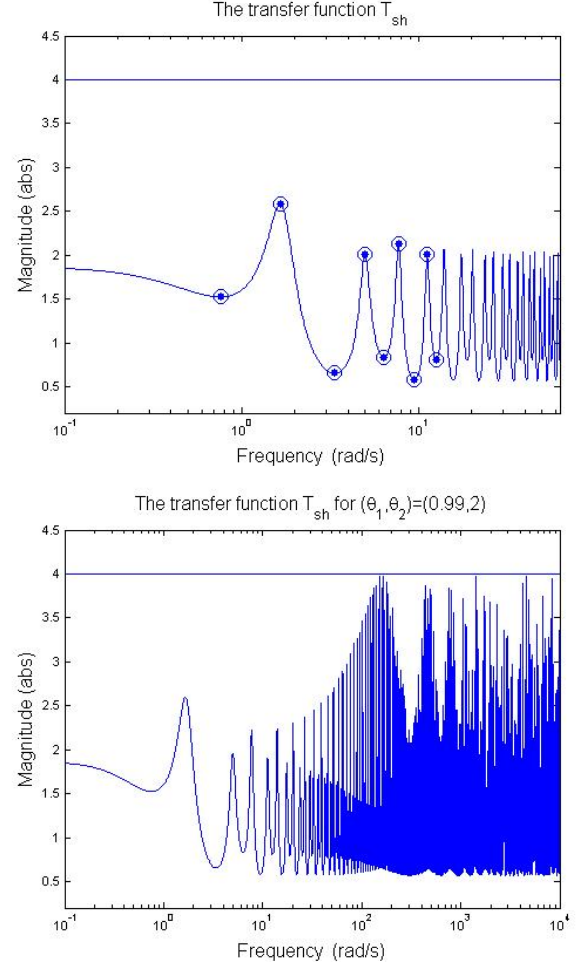


Fig. 3. (top) The magnitude Bode plot of T_{sh} and its extremum points computed at prediction and correction steps (bottom), same Bode plot for $(\tau_1, \tau_2) = (0.99, 2)$.

6 Concluding Remarks

We presented novel algorithms for the computation of extremal singular values and strong \mathcal{H}_∞ norms of SISO time-delay systems described by DDAEs. The latter algorithm does need the iteration inherent to level set methods.

The approach can be easily extended to multi-input-single-output (MISO) and single-input-multiple-output (SIMO) systems. The algorithm namely relies on the property that the singular value plot of the transfer function involves one singular value curve. Whether the presented approach can be extended to general MIMO systems is an open problem.

Acknowledgements

This work has been supported by the Belgian Federal Science Policy Office, the KU Leuven research council

and the Research Foundation-Flanders (FWO). The first author thanks Elfin D. Gumussoy for the fruitful discussions.

References

- [1] S. Boyd and V. Balakrishnan. A regularity result for the singular values of a transfer matrix and a quadratically convergent algorithm for computing its \mathcal{L}_∞ -norm. *Systems & Control Letters*, 15:1–7, 1990.
- [2] S. Boyd, V. Balakrishnan, and P. Kabamba. A bisection method for computing the \mathcal{H}_∞ norm of a transfer matrix and related problems. *Mathematics of Control, Signals and Systems*, 2:207–219, 1989.
- [3] D. Breda, S. Maset, and R. Vermiglio. Pseudospectral differencing methods for characteristic roots of delay differential equations. *SIAM Journal on Scientific Computing*, 27(2):482–495, 2005.
- [4] D. Breda, S. Maset, and R. Vermiglio. Pseudospectral approximation of eigenvalues of derivative operators with non-local boundary conditions. *Applied Numerical Mathematics*, 56(3-4):318–331, 2006.
- [5] N.A. Bruinsma and M. Steinbuch. A fast algorithm to compute the \mathcal{H}_∞ -norm of a transfer function matrix. *Systems & Control Letters*, 14:287–293, 1990.
- [6] R. Byers. A bisection method for measuring the distance of a stable matrix to the unstable matrices. *SIAM Journal on Scientific and Statistical Computing*, 9(9):875–881, 1988.
- [7] P. Gahinet and L. F. Shampine. Software for modeling and analysis of linear systems with delays. In *Proceedings of the American Control Conference*, 2004.
- [8] S. Gumussoy, B. Eryilmaz, and P. Gahinet. Working with time-delay systems in MATLAB. In *10th IFAC Workshop on Time Delay Systems*, Northeastern University, USA, June 22-24 2012.
- [9] S. Gumussoy and W. Michiels. Fixed-order H-infinity control for interconnected systems using delay differential algebraic equations. *SIAM Journal on Control and Optimization*, 49(2):2212–2238, 2011.
- [10] J.K. Hale and S.M. Verduyn Lunel. Strong stabilization of neutral functional differential equations. *IMA Journal of Mathematical Control and Information*, 19:5-23, 2002.
- [11] W. Michiels. Spectrum based stability analysis and stabilization of systems described by delay differential algebraic equations. *IET Control Theory and Applications*, 5(16):1829–1842, 2011.
- [12] W. Michiels and S. Gumussoy. Characterization and computation of H-infinity norms of time-delay systems. *SIAM Journal on Matrix Analysis and Applications*, 31(4):2093–2115, 2010.
- [13] C.W. Scherer and S. Weiland. Linear Matrix Inequalities in Control. In W.S. Levine (Ed.), *The Control Systems Handbook*, 2nd Ed., CRC Press, 2010.
- [14] S. Skogestad and I. Postlethwaite. *Multivariable Feedback Control*. John Wiley, 1996.
- [15] K. Zhou, J.C. Doyle, and K. Glover. *Robust and optimal control*. Prentice Hall, 1995.